

### 3.2 DESCRIPTIONS OF SUBROUTINES IN ALPHABETICAL ORDER

SCAL ARG5  
SCAL 60H  
RST 18H:DEFB 60H  
DF 60

ARG5 simply transfers the contents of workspace locations into CPU register pairs:

ARG1 into HL  
ARG2 into DE  
ARG3 into BC.

ARG1 is locations 0C0C/D,  
ARG2 is locations 0C0E/F,  
ARG3 is locations 0C10/1.

---

#### ADDRESS TABLE EXECUTION

SCAL ATE  
SCAL 73H  
RST 18H:DEFB 73H  
DF 73

ATE automatically calls a sequence of other routines which are listed in a table. On entry, the start address of the table is in register pair HL. The routine ends when either the Carry flag is set (the input device handlers set the Carry flag when an input is detected) or when the end of the table is reached. The table pointed to by HL comprises the numbers of the routines to be called, terminated by 00. The routines are normally input or output routines. Thus to automatically execute CRT and KBD (ie. to output to the video display and then scan the keyboard, the table would be 65H(CRT), 61H(KBD), 00(terminator).

Since facility exists within NAS-SYS for changing the input and output tables used by the input and output device drivers listed in section 3.19 and h, routine ATE will rarely need to be used explicitly.

See also commands U and X in chapter 2, and NIM, NOM, NNIM, NNOM in this section.

---

SCAL B1HEX  
SCAL 7AH  
RST 18H:DEFB 7AH  
DF 7A

B1HEX transmits the ASCII code for the hexadecimal representation of the low order nibble (ie. the right-hand four bits) of register A to the devices in the output table. The screen display shows this as a single hexadecimal character, 0 to F. The serial output port transmits the ASCII code for this character, eg. if register A contains 1AH, the byte 41H is transmitted, this being the ASCII code for character A.

```
-----
SCAL B2HEX
SCAL 68H
RST 18H:DEFB 68H
DF 68
-----
```

B2HEX transmits the ASCII codes for the hexadecimal representation of the contents of register A to the devices in the output table. The screen display shows the contents of register A as two hexadecimal digits. The serial output port transmits the ASCII code for the two characters, eg. if register A contains 4AH, the bytes 34H and 41H are transmitted, these being the ASCII codes for characters 4 and A respectively.

```
-----
SCAL BLINK
SCAL 7BH
RST 18H:DEFB 7BH
DF 7B
-----
```

BLINK waits for an input from the devices in the input table, (normally the keyboard and the serial input port) returning only when an input is received. While waiting for an input, the cursor on the screen is blinked. The routine returns with the ASCII code for the input character in register A.

NAS-SYS 3 allows the blink rate of the cursor to be set by the user. Workspace location KBLINK (0C32/3), which is initialised to 0100H, may be altered.

```
-----
RST BRKPT
RST 20H
E7
-----
```

BRKPT copies the Z80 cpu registers into the register save area of the workspace, displays them, and then returns control to NAS-SYS. The register display format is the same as that described under 'command P', chapter 2.

```
-----
SCAL CP0S
SCAL 7CH
RST 18H:DEFB 7CH
DF 7C
-----
```

On entry register pair HL points to a position on the video display. On exit, HL points to the start of that line on the screen.



```
SCAL CRLF
SCAL 6AH
RST 18H:DEFB 6AH
DF 6A
```

CRLF transmits the byte 0DH (ASCII code for carriage return) to all the devices in the output table. The screen display routine moves the cursor to the beginning of the next line. The routine simply loads register A with 0DH (the newline code) and then calls ROUT.

```
SCAL CRT
SCAL 65H
RST 18H:DEFB 65H
DF 65
```

CRT is the video display handler. It writes the character whose ASCII code is in register A onto the screen at the current cursor position. The character may be a screen edit command.

```
eg 1      3E 41      LD A,41H  Write A on screen at
          DF 65      SCAL CRT  current cursor position.
```

```
eg 2      3E 07      LD A,07H  Write BEL character at
          DF 65      SCAL CRT  current cursor position.
```

```
SCAL ERRM
SCAL 6BH
RST 18H:DEFB 6BH
DF 6B
```

ERRM transmits the bytes 45H, 72H, 72H, 6FH, 72H (the ASCII codes for 'Error') to all the devices in the output table; the code for a carriage return/line feed follows the Error message. The screen display shows the message 'Error' at the current cursor position. The routine is simply a call to PRS with the ASCII codes for Error as the string, followed by a call to CRLF.

```
SCAL FFLP
SCAL 5EH
RST 18H:DEFB 5EH
DF 5E
```

FFLP changes the state of one or more bits of output port 0, then immediately restores them to their original state. The bits that are changed are the same as those bits of register A which contain a 1. For example, if register A contains 24H or 00100100 then bits 2 and 5 will be pulsed.

Most of the bits of output port 0 are used to drive the keyboard; before using this command refer to Chapter 1.1.